

Degree reduction of composite Bézier curves

Przemysław Gospodarczyk*, Stanisław Lewanowicz, Paweł Woźny

Institute of Computer Science, University of Wrocław, ul. Joliot-Curie 15, 50-383 Wrocław, Poland

Abstract

This paper deals with the problem of multi-degree reduction of a composite Bézier curve with the parametric continuity constraints at the endpoints of the segments. We present a novel method which is based on the idea of using constrained dual Bernstein polynomials to compute the control points of the reduced composite curve. In contrast to other methods, ours minimizes the L_2 -error for the whole composite curve instead of minimizing the L_2 -errors for each segment separately. As a result, an additional optimization is possible. Examples show that the new method gives much better results than multiple application of the degree reduction of a single Bézier curve.

Keywords: Composite Bézier curve, multi-degree reduction, parametric continuity constraints, least-squares approximation, constrained dual Bernstein basis.

1. Introduction

In recent years, many methods have been used to reduce the degree of Bézier curves with constraints (see, e.g., [1–6, 8, 9, 11, 13–20]). Most of these papers give methods of multi-degree reduction of a *single* Bézier curve with constraints of endpoints (parametric or geometric) continuity of arbitrary order with respect to L_2 -norm. Observe, however, that degree reduction schemes often need to be combined with the subdivision algorithm, i.e., a high degree curve is replaced by a number of lower degree curve segments, or a *composite* Bézier curve, and continuity between adjacent lower degree curve segments should be maintained. Intuitively, a possible approach in such a case is applying the multi-degree reduction procedure to one segment of the curve after another with properly chosen endpoints continuity constraints. However, in general, the obtained solution does not minimize the distance between two composite curves.

In this paper, we give the optimal least-squares solution of multi-degree reduction of a composite Bézier curve with the parametric continuity constraints at the endpoints of the segments. More specifically, we consider the following approximation problem.

Problem 1.1 (*C^r -constrained multi-degree reduction of the composite Bézier curve*). Let $a = t_0 < t_1 < \dots < t_s = b$ be a partition of the interval $[a, b]$. Let there be given a degree $\mathbf{n} = (n_1, n_2, \dots, n_s)$ composite Bézier curve $P(t)$ ($t \in [a, b]$) in \mathbb{R}^d that in the interval $[t_{i-1}, t_i]$ ($i = 1, 2, \dots, s$) is exactly represented as a Bézier curve $P_i(u)$ ($u \in [0, 1]$) of degree n_i , i.e.,

$$P(t) = P_i(u) := \sum_{j=0}^{n_i} p_{i,j} B_j^{n_i}(u) \quad (t_{i-1} \leq t \leq t_i; u := (t - t_{i-1})/h_i),$$

where $h_i := t_i - t_{i-1}$, and

$$B_j^n(u) := \binom{n}{j} u^j (1-u)^{n-j} \quad (0 \leq j \leq n)$$

*Corresponding author. Fax +48 71 3757801

Email addresses: pgo@ii.uni.wroc.pl (Przemysław Gospodarczyk), Stanislaw.Lewanowicz@ii.uni.wroc.pl (Stanisław Lewanowicz), Pawel.Wozny@ii.uni.wroc.pl (Paweł Woźny)

are the Bernstein basis polynomials of degree n . Find a composite Bézier curve $Q(t)$ ($t \in [a, b]$) of degree $\mathbf{m} = (m_1, m_2, \dots, m_s)$ which in the interval $[t_{i-1}, t_i]$ ($i = 1, 2, \dots, s$) is exactly represented as a Bézier curve $Q_i(u)$ ($u \in [0, 1]$) of degree $m_i < n_i$, i.e.,

$$Q(t) = Q_i(u) := \sum_{j=0}^{m_i} q_{i,j} B_j^{m_i}(u) \quad (t_{i-1} \leq t \leq t_i; u := (t - t_{i-1})/h_i), \quad (1.1)$$

such that the squared L_2 -error

$$E := \int_a^b \|P(t) - Q(t)\|^2 dt = \sum_{i=1}^s E_i, \quad (1.2)$$

where

$$E_i := \int_{t_{i-1}}^{t_i} \|P(t) - Q(t)\|^2 dt = h_i \int_0^1 \|P_i(u) - Q_i(u)\|^2 du,$$

reaches the minimum under the additional conditions that

$$\left. \frac{d^j Q(t)}{dt^j} \right|_{t=t_0} = \left. \frac{d^j P(t)}{dt^j} \right|_{t=t_0} \quad (j = 0, 1, \dots, r_0), \quad (1.3)$$

$$\left. \frac{d^j Q(t)}{dt^j} \right|_{t=t_i-} = \left. \frac{d^j Q(t)}{dt^j} \right|_{t=t_i+} \quad (i = 1, 2, \dots, s-1; j = 0, 1, \dots, r_i), \quad (1.4)$$

$$\left. \frac{d^j Q(t)}{dt^j} \right|_{t=t_s} = \left. \frac{d^j P(t)}{dt^j} \right|_{t=t_s} \quad (j = 0, 1, \dots, r_s), \quad (1.5)$$

where $r_j \geq 0$ ($j = 0, 1, \dots, s$) and $r_{i-1} + r_i < m_i - 1$ ($i = 1, 2, \dots, s$). We will say that the curves P and Q satisfy the $C^{\mathbf{r}}$ -continuity conditions at the points t_0, t_1, \dots, t_s , where we use the notation $\mathbf{r} := (r_0, r_1, \dots, r_s)$. Here $\|\cdot\|$ is the Euclidean vector norm.

Remark 1.2. One may think that the conditions

$$\left. \frac{d^j Q(t)}{dt^j} \right|_{t=t_i} = \left. \frac{d^j P(t)}{dt^j} \right|_{t=t_i} \quad (i = 1, 2, \dots, s-1; j = 0, 1, \dots, r_i)$$

would be more natural than (1.4). However, in contrast to our new method, such an approach leaves no room for additional optimization.

Remark 1.3. Sometimes, it may be useful to interpolate the endpoints of the original segments (see Example 4.2), i.e., to demand that $Q(t_i) = P(t_i)$ holds for $i = 1, 2, \dots, s-1$. In such a case, constraints (1.4) should be appropriately modified by restricting the range of j to $1, 2, \dots, r_i$.

The paper is organized as follows. In Section 2, we recall some results which are later applied in the solution of the problem, given in Section 3. Several illustrative examples are presented in Section 4. Finally, Section 5 contains some concluding remarks.

We end this section with introducing some notation. The *shifted factorial* is defined by $(c)_0 := 1$, $(c)_j := c(c+1) \cdots (c+j-1)$ ($j = 1, 2, \dots$). The *iterated forward difference operator* Δ^j is given by

$$\Delta^j \alpha_k := \Delta^{j-1} \alpha_{k+1} - \Delta^{j-1} \alpha_k \quad (j = 1, 2, \dots) \quad \text{and} \quad \Delta^0 \alpha_k := \alpha_k.$$

Moreover, we adopt the convention that in an expression of the form $\Delta^j \gamma_{i,k}$ the operator Δ^j acts on the second variable (first variable being fixed), e.g.,

$$\Delta^2 \gamma_{i,k} = \Delta \gamma_{i,k+1} - \Delta \gamma_{i,k} = \gamma_{i,k+2} - 2\gamma_{i,k+1} + \gamma_{i,k}.$$

2. Preliminaries

Let us denote by $\Pi_m^{(k,l)} = \text{span}\{B_{k+1}^m, B_{k+2}^m, \dots, B_{m-l-1}^m\}$, where k and l are natural numbers such that $k+l < m-1$, the space of all polynomials of degree at most m , whose derivatives of order $\leq k$ at $t=0$ and of order $\leq l$ at $t=1$ vanish. There is a unique *dual constrained Bernstein basis of degree m* (see, e.g., [10]), $D_{k+1}^{(m,k,l)}, D_{k+2}^{(m,k,l)}, \dots, D_{m-l-1}^{(m,k,l)}$, satisfying $\langle D_j^{(m,k,l)}, B_h^m \rangle = \delta_{j,h}$ ($j, h = k+1, k+2, \dots, m-l-1$), where the inner product $\langle \cdot, \cdot \rangle$ is given by $\langle f, g \rangle := \int_0^1 f(t)g(t) dt$, $\delta_{j,h}$ equals 1 if $j = h$, and 0 otherwise. The table of the coefficients $C_{j,h} \equiv C_{j,h}(m, k, l)$ in the connection formulas

$$D_j^{(m,k,l)}(x) = \sum_{h=k+1}^{m-l-1} C_{j,h}(m, k, l) B_h^m(x) \quad (j = k+1, k+2, \dots, m-l-1) \quad (2.1)$$

can be efficiently computed using the following algorithm (by convention, $C_{j,h} := 0$ if j or $h \notin \{k+1, \dots, m-l-1\}$):

Algorithm 2.1 (Computing the C -table [12]).

1. Compute the entries of the first row of the table by

$$C_{k+1, m-l-1} := \binom{m}{k+1}^{-1} \binom{m}{l+1}^{-1} \frac{(-1)^{m-k-l-2} (m+k+l+3)!}{(m-k-l-2)!(2k+2)!(2l+2)!},$$

$$C_{k+1, h} := \frac{(h-m)(h-k)(h+k+3)}{(h+1)[(h-m)^2 - (l+1)^2]} C_{k+1, h+1} \quad (h = m-l-2, \dots, k+2, k+1).$$

2. For $j = k+1, k+2, \dots, m-l-2$, compute

$$C_{j+1, h} = \frac{1}{A(j)} \{2(j-h)(j+h-m) C_{j, h} + B(h) C_{j, h-1} + A(h) C_{j, h+1} - B(j) C_{j-1, h}\}$$

$$(h = k+1, k+2, \dots, m-l-1)$$

with $A(u) := (u-m)(u-k)(u+k+2)/(u+1)$, $B(u) := u(u-m-l-2)(u-m+l)/(u-m-1)$.

Remark 2.2. Note that the complexity of the algorithm is $O(m^2)$. In [14, Proposition 3], Lu observed that—due to some symmetries in the C -table—only part of the entries must be calculated as in Algorithm 2.1.

In the solution of Problem 1.1 given in Section 3, we need the following lemma which follows easily from the results obtained in [12, Thm 4.1] and [20, Thm 4.1].

Lemma 2.3 (Multi-degree reduction of a Bézier curve with prescribed boundary control points). *Let there be given natural numbers n, m, k, l such that $m < n$ and $k+l < m-1$. Let P^* be the Bézier curve of degree n ,*

$$P^*(u) := \sum_{j=0}^n p_j B_j^n(u) \quad (0 \leq u \leq 1).$$

The Bézier curve of degree m ,

$$Q^*(u) := \sum_{j=0}^m q_j B_j^m(u) \quad (0 \leq u \leq 1),$$

having the prescribed control points q_0, q_1, \dots, q_k and $q_{m-l}, q_{m-l+1}, \dots, q_m$, and the inner control points

$$q_j := \sum_{h=k+1}^{m-l-1} C_{j,h}(m, k, l) v_h - \left(\sum_{h=0}^k + \sum_{h=m-l}^m \right) K_{j,h} q_h \quad (j = k+1, k+2, \dots, m-l-1),$$

where

$$v_h := (n+m+1)^{-1} \binom{m}{h} \sum_{i=0}^n \binom{n}{i} \binom{n+m}{i+h}^{-1} p_i,$$

$$K_{j,h} := \binom{m}{h} \binom{m}{j}^{-1} \frac{(-1)^{j-k-1} (k+1-h)_{m-k-l-1} (k+2)_h (l+2)_{m-h}}{(j-h)(j-k-1)!(m-l-j-1)!(k+2)_j (l+2)_{m-j}},$$

and $C_{j,h}(m, k, l)$ are the coefficients in (2.1), gives the least value of the squared L_2 -error

$$E^* := \int_0^1 \|P^*(u) - Q^*(u)\|^2 du,$$

equal to

$$E^* = \sum_{h=1}^d [I_{n,n}(\mathbf{p}^h, \mathbf{p}^h) + I_{m,m}(\mathbf{q}^h, \mathbf{q}^h) - 2I_{n,m}(\mathbf{p}^h, \mathbf{q}^h)],$$

where we use the notation $\mathbf{p}^h, \mathbf{q}^h$ for the vectors of h th coordinates of points p_0, p_1, \dots, p_n and q_0, q_1, \dots, q_m , respectively, and where for $\mathbf{x} := [x_0, x_1, \dots, x_N]$ and $\mathbf{y} := [y_0, y_1, \dots, y_M]$, we define

$$I_{N,M}(\mathbf{x}, \mathbf{y}) := (N+M+1)^{-1} \sum_{i=0}^N \sum_{j=0}^M \binom{N+M}{i+j}^{-1} \binom{N}{i} \binom{M}{j} x_i y_j.$$

3. Main result

Recall that for an arbitrary Bézier curve of degree N ,

$$W_N(u) = \sum_{h=0}^N w_h B_h^N(u) \quad (0 \leq u \leq 1),$$

the following well-known formulas hold (see, e.g., [7, §5.3]):

$$\frac{d^j W_N(u)}{du^j} \Big|_{u=0} = (N-j+1)_j \Delta^j w_0 = (N-j+1)_j \sum_{h=0}^j (-1)^{j+h} \binom{j}{h} w_h,$$

$$\frac{d^j W_N(u)}{du^j} \Big|_{u=1} = (N-j+1)_j \Delta^j w_{N-j} = (N-j+1)_j \sum_{h=0}^j (-1)^{j+h} \binom{j}{h} w_{N-j+h}.$$

Moreover, notice that

$$\begin{aligned} \frac{d^j P(t)}{dt^j} \Big|_{t=t_0} &= h_1^{-j} \frac{d^j P_1(u)}{du^j} \Big|_{u=0}, \\ \frac{d^j P(t)}{dt^j} \Big|_{t=t_s} &= h_s^{-j} \frac{d^j P_s(u)}{du^j} \Big|_{u=1}, \\ \frac{d^j Q(t)}{dt^j} \Big|_{t=t_i+} &= h_{i+1}^{-j} \frac{d^j Q_{i+1}(u)}{du^j} \Big|_{u=0} \quad (i = 0, 1, \dots, s-1), \\ \frac{d^j Q(t)}{dt^j} \Big|_{t=t_i-} &= h_i^{-j} \frac{d^j Q_i(u)}{du^j} \Big|_{u=1} \quad (i = 1, 2, \dots, s), \end{aligned}$$

where the notation used is that of Problem 1.1. Using the above equations in (1.3) and (1.5), we obtain

$$q_{1,j} = \binom{n_1}{j} \binom{m_1}{j}^{-1} \Delta^j p_{1,0} - \sum_{h=0}^{j-1} (-1)^{j+h} \binom{j}{h} q_{1,h} \quad (j = 0, 1, \dots, r_0), \quad (3.1)$$

$$q_{s,m_s-j} = (-1)^j \binom{n_s}{j} \binom{m_s}{j}^{-1} \Delta^j p_{s,n_s-j} - \sum_{h=1}^j (-1)^h \binom{j}{h} q_{s,m_s-j+h} \quad (j = 0, 1, \dots, r_s). \quad (3.2)$$

Similarly, conditions (1.4) imply that for $i \in \{1, 2, \dots, s-1\}$, we have

$$\left. \begin{aligned} q_{i,m_i-h} &= \kappa_i + \sum_{j=1}^h (-1)^j \binom{h}{j} a_{i,j} \lambda_{i,j}, \\ q_{i+1,h} &= \kappa_i + \sum_{j=1}^h \binom{h}{j} a_{i+1,j} \lambda_{i,j} \end{aligned} \right\} \quad (h = 0, 1, \dots, r_i), \quad (3.3)$$

where

$$a_{i,j} := \frac{h_i^j}{(m_i - j + 1)_j},$$

for an arbitrary point $\kappa_i = (\kappa_i^1, \kappa_i^2, \dots, \kappa_i^d)$ and vectors $\lambda_{i,j} = [\lambda_{i,j}^1, \lambda_{i,j}^2, \dots, \lambda_{i,j}^d]$. In particular, we have

$$\begin{aligned} q_{i,m_i} &= \kappa_i, & q_{i,m_i-1} &= \kappa_i - a_{i,1} \lambda_{i,1}, \\ q_{i+1,0} &= \kappa_i, & q_{i+1,1} &= \kappa_i + a_{i+1,1} \lambda_{i,1}, \\ q_{i,m_i-2} &= \kappa_i - 2a_{i,1} \lambda_{i,1} + a_{i,2} \lambda_{i,2}, \\ q_{i+1,2} &= \kappa_i + 2a_{i+1,1} \lambda_{i,1} + a_{i+1,2} \lambda_{i,2}, \\ q_{i,m_i-3} &= \kappa_i - 3a_{i,1} \lambda_{i,1} + 3a_{i,2} \lambda_{i,2} - a_{i,3} \lambda_{i,3}, \\ q_{i+1,3} &= \kappa_i + 3a_{i+1,1} \lambda_{i,1} + 3a_{i+1,2} \lambda_{i,2} + a_{i+1,3} \lambda_{i,3}. \end{aligned}$$

Coming back to the problem of constrained multi-degree reduction of a composite Bézier curve (see Problem 1.1), let us observe that for any $i \in \{1, 2, \dots, s\}$ the formulas (3.1)–(3.3) with *fixed parameters* κ_i and $\lambda_{i,j}$ constitute constraints of the form demanded in Lemma 2.3. Now, by applying this lemma for $i = 1, 2, \dots, s$ to $P^* := P_i$, $Q^* := Q_i$ with $n := n_i$, $m := m_i$, $k := r_{i-1}$ and $l := r_i$, we obtain the set of the segments of the composite Bézier curve Q (1.1) with control points depending on the parameters. In order to solve Problem 1.1, we have to determine the *optimum values* of the parameters (cf. Remark 1.2).

Let us denote

$$\mathbf{\Lambda}_i := [\kappa_i, \lambda_{i,1}, \lambda_{i,2}, \dots, \lambda_{i,r_i}]^T = [\kappa_i^1, \dots, \kappa_i^d, \lambda_{i,1}^1, \dots, \lambda_{i,1}^d; \dots; \lambda_{i,r_i}^1, \dots, \lambda_{i,r_i}^d]^T \in \mathbb{R}^{\rho_i},$$

where $\rho_i := (r_i + 1)d$. The *optimum values* of the parameters can be obtained by minimizing the error function (1.2),

$$\begin{aligned} E &\equiv E(\mathbf{\Lambda}_1; \dots; \mathbf{\Lambda}_{s-1}) \\ &= E_1(\mathbf{\Lambda}_1) + \sum_{i=2}^{s-1} E_i(\mathbf{\Lambda}_{i-1}; \mathbf{\Lambda}_i) + E_s(\mathbf{\Lambda}_{s-1}), \end{aligned} \quad (3.4)$$

where

$$E_i = h_i \sum_{h=1}^d [I_{n_i, n_i}(\mathbf{p}_i^h, \mathbf{p}_i^h) + I_{m_i, m_i}(\mathbf{q}_i^h, \mathbf{q}_i^h) - 2I_{n_i, m_i}(\mathbf{p}_i^h, \mathbf{q}_i^h)], \quad (3.5)$$

$E_1 \equiv E_1(\mathbf{\Lambda}_1)$, $E_i \equiv E_i(\mathbf{\Lambda}_{i-1}; \mathbf{\Lambda}_i)$ ($i = 2, 3, \dots, s-1$), $E_s \equiv E_s(\mathbf{\Lambda}_{s-1})$, with $\mathbf{p}_i^h, \mathbf{q}_i^h$ being the vectors of h th coordinates of points $p_{i,0}, p_{i,1}, \dots, p_{i,n_i}$ and $q_{i,0}, q_{i,1}, \dots, q_{i,m_i}$, respectively. For a minimum of E , it is necessary that the derivatives of E with respect to the parameters κ_i^h and $\lambda_{i,j}^h$ are zero, which yields a system of σ linear equations with σ unknowns, where $\sigma := \rho_1 + \rho_2 + \dots + \rho_{s-1}$. Hence, for $i = 1, 2, \dots, s-1$, we have

$$\frac{\partial E}{\partial \kappa_i^h} = \frac{\partial}{\partial \kappa_i^h}(E_i + E_{i+1}) = 0 \quad (h = 1, 2, \dots, d), \quad (3.6)$$

$$\frac{\partial E}{\partial \lambda_{i,j}^h} = \frac{\partial}{\partial \lambda_{i,j}^h}(E_i + E_{i+1}) = 0 \quad (j = 1, 2, \dots, r_i; h = 1, 2, \dots, d). \quad (3.7)$$

Now, we summarise the whole idea in Algorithm 3.1. Note that for $i = 1, 2, \dots, s$, the use of Lemma 2.3 requires computation of C -table (see Algorithm 2.1). The entries of i th C -table are denoted by $C_{j,h}^{(i)}$ ($j, h = r_{i-1} + 1, r_{i-1} + 2, \dots, m_i - r_i - 1$).

Algorithm 3.1 (C^r -constrained multi-degree reduction of composite Bézier curves).

Input: $s, n_i, m_i, p_{i,0}, p_{i,1}, \dots, p_{i,n_i}$ ($i = 1, 2, \dots, s$),

$r_0, r_1, \dots, r_s, a = t_0 < t_1 < \dots < t_s = b$

Output: solution $q_{i,j}$ ($i = 1, 2, \dots, s; j = 0, 1, \dots, m_i$) of Problem 1.1

Step 1. For $i = 1, 2, \dots, s$, compute $C_{j,h}^{(i)}$ ($j, h = r_{i-1} + 1, r_{i-1} + 2, \dots, m_i - r_i - 1$) using Algorithm 2.1, assuming that $m := m_i, k := r_{i-1}, l := r_i$.

Step 2. Compute E_i ($i = 1, 2, \dots, s$) by (3.5).

Step 3. Solve the system of linear equations (3.6), (3.7).

Step 4. Compute $q_{1,j}$ ($j = 0, 1, \dots, r_0$) by (3.1).

Step 5. Compute q_{s,m_s-j} ($j = 0, 1, \dots, r_s$) by (3.2).

Step 6. Compute q_{i,m_i-h} and $q_{i+1,h}$ ($i = 1, 2, \dots, s-1; h = 0, 1, \dots, r_i$) by (3.3).

Step 7. For $i = 1, 2, \dots, s$, set $P^* := P_i, Q^* := Q_i, n := n_i, m := m_i, k := r_{i-1}, l := r_i$, and compute the control points $q_{i,j}$ ($j = r_{i-1} + 1, r_{i-1} + 2, \dots, m_i - r_i - 1$) using Lemma 2.3.

Step 8. Return $q_{i,j}$ ($i = 1, 2, \dots, s; j = 0, 1, \dots, m_i$).

Remark 3.2. In case, where interpolation conditions are imposed at the endpoints of the original segments (see Remark 1.3), Algorithm 3.1 must be slightly modified. Note that the parameter κ_i ($i = 1, 2, \dots, s-1$) is the meeting point of the consecutive segments Q_i and Q_{i+1} , i.e., $\kappa_i = Q_i(1) = Q_{i+1}(0)$. Therefore, by setting $\kappa_i := P(t_i)$ (cf. (3.3)), and by removing the subsystem (3.6) from the system, the goal is easily achieved.

4. Examples

This section provides of the application of our algorithm. We give the squared L_2 -errors E_i ($i = 1, 2, \dots, s$) and E (see (3.5), (3.4)) as well as the maximum errors

$$E_i^\infty := \max_{t \in U_i} \|P(t) - Q(t)\| \approx \max_{t \in [t_{i-1}, t_i]} \|P(t) - Q(t)\|,$$

$$E^\infty := \max_{1 \leq i \leq s} E_i^\infty,$$

where $U_i := \{t_{i-1}, t_{i-1} + \delta, t_{i-1} + 2\delta, \dots, t_i\}$ with $\delta := (t_i - t_{i-1})/500$.

Results of the experiments have been obtained in MapleTM13, using 32-digit arithmetic. The system of linear equations (3.6), (3.7) is solved using MapleTM `fsolve` procedure.

Example 4.1. Assuming that $s = 4$ and $t_0 = 0$, $t_1 = 0.18$, $t_2 = 0.29$, $t_3 = 0.53$, $t_4 = 1$, the composite curve “Squirrel” is formed by four Bézier segments of degrees 9, 12, 12, and 12, respectively. Control points are given at <http://www.ii.uni.wroc.pl/~pgo/squirrel.txt>. For the results of degree reduction, see Table 1 as well as the corresponding Figures 1a and 1b. This example shows that Algorithm 3.1 may result in a lower error than multiple application of [20, Algorithm 1]. Furthermore, the larger are r_i ’s, the bigger are differences in errors because we have more degrees of freedom, i.e., parameters $\lambda_{i,j}$ (see Remark 1.2).

Parameters	i	Algorithm 3.1		[20, Algorithm 1]	
		E_i	E_i^∞	E_i	E_i^∞
$\mathbf{m} = (7, 8, 8, 9)$	1	$3.40e-8$	$1.64e-3$	$3.40e-8$	$1.01e-3$
	2	$4.98e-7$	$5.20e-3$	$7.82e-7$	$5.36e-3$
$\mathbf{r} = (2, 0, 3, 2, 2)$	3	$1.85e-6$	$8.23e-3$	$7.58e-6$	$1.13e-2$
	4	$1.28e-6$	$3.02e-3$	$3.24e-6$	$5.07e-3$
Summary		E	E^∞	E	E^∞
		$3.66e-6$	$8.23e-3$	$1.16e-5$	$1.13e-2$
$\mathbf{m} = (7, 8, 8, 9)$	1	$9.62e-8$	$1.62e-3$	$1.10e-7$	$1.84e-3$
	2	$7.26e-7$	$6.26e-3$	$1.14e-5$	$1.95e-2$
$\mathbf{r} = (3, 0, 4, 2, 3)$	3	$3.28e-6$	$9.80e-3$	$6.81e-4$	$9.51e-2$
	4	$3.03e-6$	$4.34e-3$	$9.37e-6$	$8.24e-3$
Summary		E	E^∞	E	E^∞
		$7.13e-6$	$9.80e-3$	$7.02e-4$	$9.51e-2$

Table 1: Squared L_2 -errors and maximum errors for degree reduction of the composite Bézier curve “Squirrel”.

Example 4.2. For $s = 2$ and $t_0 = 0$, $t_1 = 0.49$, $t_2 = 1$, we consider the composite curve “L” which consists of two Bézier segments of degrees 8 and 12 having the control points $\{(0.313, 0.52), (0.198, 0.493), (0.245, 0.412), (0.346, 0.446), (0.466, 0.528), (0.397, 0.518), (0.301, 0.553), (0.296, 0.473), (0.299, 0.418)\}$, and $\{(0.299, 0.418), (0.301, 0.38), (0.323, 0.342), (0.328, 0.294), (0.256, 0.252), (0.23, 0.272), (0.173, 0.323), (0.237, 0.427), (0.294, 0.278), (0.35, 0.327), (0.4, 0.267), (0.417, 0.296), (0.396, 0.323)\}$, respectively. The comparison of the results of Algorithm 3.1, algorithm described in Remark 3.2, and [20, Algorithm 1] is given in Table 2 (see also Figure 2). Once again, we observe that the new approach may lead to better results than the older one. Moreover, we note that in some cases it is useful to interpolate the endpoints of the original segments (see Remarks 1.3 and 3.2).

Parameters	i	Algorithm 3.1		Remark 3.2		[20, Algorithm 1]	
		E_i	E_i^∞	E_i	E_i^∞	E_i	E_i^∞
$\mathbf{m} = (6, 7)$	1	$1.00e-6$	$3.98e-3$	$1.23e-6$	$3.10e-3$	$4.74e-5$	$1.58e-2$
$\mathbf{r} = (1, 3, 1)$	2	$2.51e-6$	$3.99e-3$	$4.33e-6$	$5.49e-3$	$1.91e-5$	$1.08e-2$
Summary		E	E^∞	E	E^∞	E	E^∞
		$3.51e-6$	$3.99e-3$	$5.56e-6$	$5.49e-3$	$6.65e-5$	$1.58e-2$

Table 2: Squared L_2 -errors and maximum errors for degree reduction of the composite Bézier curve “L”.

Example 4.3. Let there be given three Bézier curves of degrees 8, 6, and 6, defined by the control points $\{(0.313, 0.52), (0.198, 0.493), (0.245, 0.412), (0.346, 0.446), (0.466, 0.528), (0.397, 0.518), (0.301, 0.553), (0.296, 0.473), (0.3, 0.422)\}$, $\{(0.305, 0.418), (0.308, 0.344), (0.408, 0.342), (0.415, 0.445), (0.405, 0.417), (0.402, 0.377), (0.4, 0.365)\}$, and $\{(0.403, 0.36), (0.395, 0.249), (0.372, 0.233),$

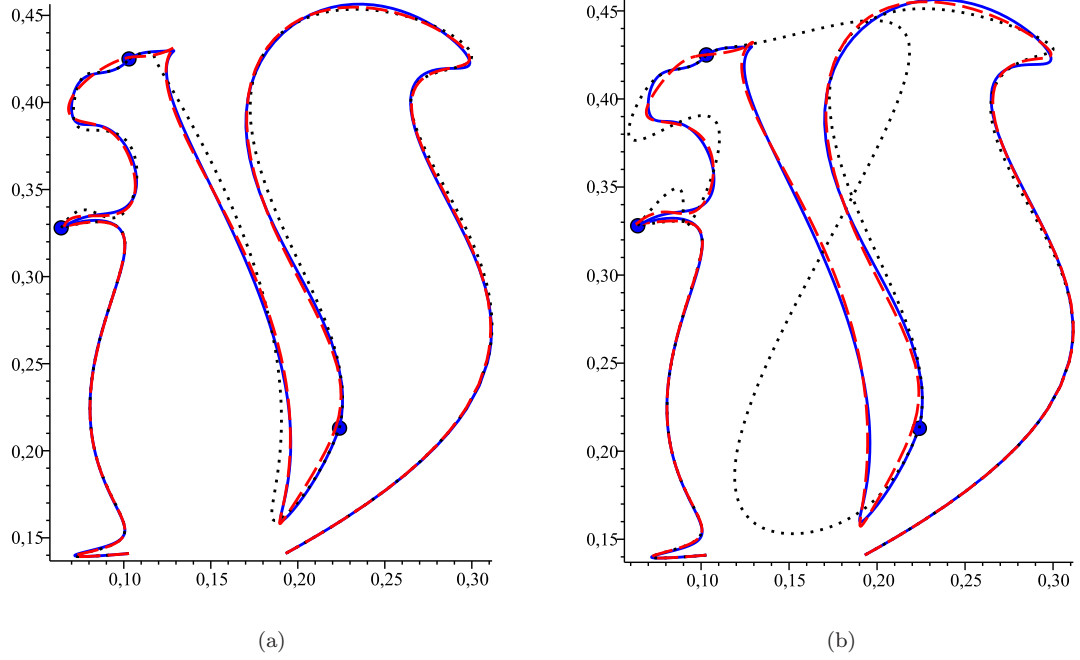


Figure 1: Original composite Bézier curve “Squirrel” of degree $\mathbf{n} = (9, 12, 12, 12)$ (blue solid line) and degree reduced composite Bézier curves computed using Algorithm 3.1 (red dashed line), and [20, Algorithm 1] (black dotted line). Parameters: (a) $\mathbf{m} = (7, 8, 8, 9)$, $\mathbf{r} = (2, 0, 3, 2, 2)$; and (b) $\mathbf{m} = (7, 8, 8, 9)$, $\mathbf{r} = (3, 0, 4, 2, 3)$.

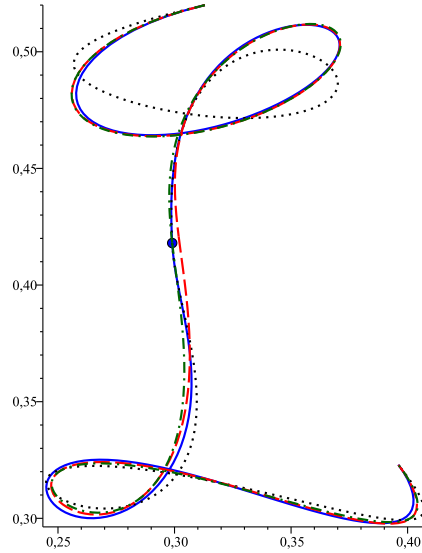


Figure 2: Original composite Bézier curve “L” of degree $\mathbf{n} = (8, 12)$ (blue solid line) and degree reduced composite Bézier curves computed using Algorithm 3.1 (red dashed line), algorithm described in Remark 3.2 (green dash-dotted line), and [20, Algorithm 1] (black dotted line). Parameters are specified in Table 2.

$(0.225, 0.228)$, $(0.302, 0.306)$, $(0.297, 0.308)$, $(0.311, 0.322)\}$, respectively. Note that these Bézier curves are not joined (see Figure 3a). In spite of that, we set $t_0 = 0$, $t_1 = 0.45$, $t_2 = 0.68$, $t_3 = 1$,

and apply Algorithm 3.1 with $\mathbf{m} = (6, 5, 5)$ and $\mathbf{r} = \mathbf{1} := (1, 1, 1, 1)$. As a result, we obtain the C^1 -continuous composite Bézier curve “G” illustrated in Figures 3b and 3c (errors: $E_1 = 9.94e-7$, $E_2 = 2.84e-6$, $E_3 = 1.42e-6$, $E = 5.25e-6$, $E_1^\infty = 1.06e-2$, $E_2^\infty = 1.42e-2$, $E_3^\infty = 9.11e-3$, $E^\infty = 1.42e-2$). In addition, the degrees of the segments are reduced. This example shows that our algorithm can serve as a tool for merging of several unconnected Bézier curves into a smooth composite Bézier curve. Furthermore, in case of C^0 -continuous input curves, the algorithm can eliminate possible *rough edges* and *corners*.

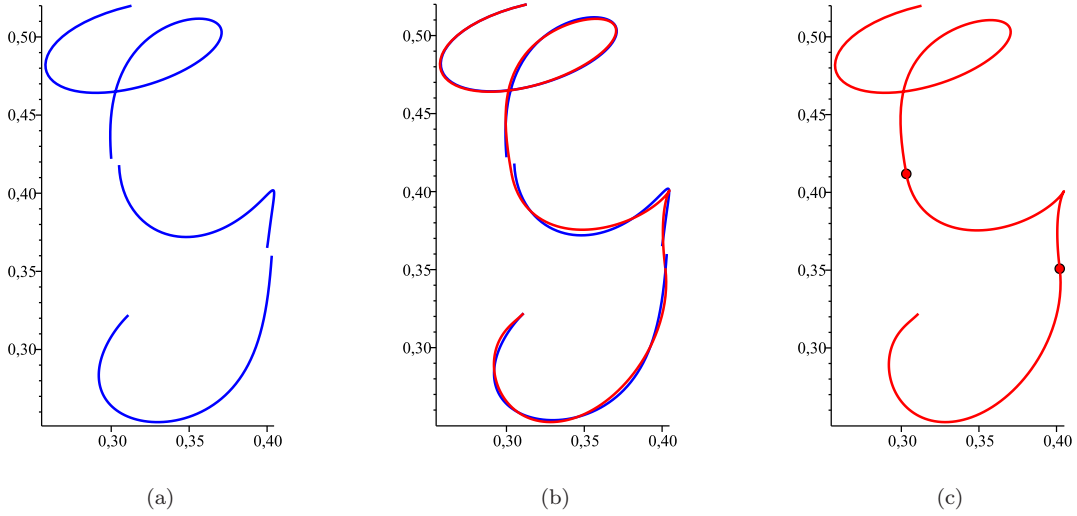


Figure 3: Three original Bézier curves of degrees 8, 6, and 6 (blue solid line) and the C^1 -continuous composite Bézier curve “G” of degree $\mathbf{m} = (6, 5, 5)$ computed using Algorithm 3.1 (red solid line) with $\mathbf{r} = \mathbf{1} := (1, 1, 1, 1)$.

5. Conclusions

We propose a novel approach to the problem of multi-degree reduction of composite Bézier curves. In contrast to other methods, ours minimizes the L_2 -error for the whole composite curve instead of minimizing the L_2 -errors for each segment separately. The main idea is to connect consecutive segments of the searched composite curve using the conditions (1.4). As a result, an additional optimization is possible. The new problem is solved efficiently using the properties of constrained dual Bernstein polynomial basis. Examples 4.1 and 4.2 show that the new method gives much better results than multiple application of the degree reduction of a single Bézier curve. Furthermore, we observe that a slight modification of the method allows for interpolation of endpoints of the original segments (see Remarks 1.3 and 3.2), which may be useful in some cases (see Example 4.2). Moreover, merging of several unconnected Bézier curves into a smooth composite Bézier curve is also possible (see Example 4.3).

Let us mention that we have studied also the extended version of Problem 1.1 where the parametric continuity conditions (1.3)–(1.5) are replaced by geometric continuity constraints. We have observed that in this case the error function E becomes a high degree polynomial function of many variables, even for modest values of r_i ’s. Consequently, we have to deal with constrained nonlinear programming problem in order to find optimal values for the parameters. Experiments show that calculations may be painfully long. So far, we have not been able to give an efficient algorithm of solving this task.

References

- [1] Y.J. Ahn, Using Jacobi polynomials for degree reduction of Bézier curves with C^k -constraints, *Comput. Aided Geom. Des.* 20 (2003), 423–434.
- [2] Y.J. Ahn, B.-G. Lee, Y. Park, J. Yoo, Constrained polynomial degree reduction in the L_2 -norm equals best weighted Euclidean approximation of Bézier coefficients, *Comput. Aided Geom. Des.* 21 (2004), 181–191.
- [3] R. Ait-Haddou, M. Bartoň, Constrained multi-degree reduction with respect to Jacobi norms, *Comput. Aided Geom. Des.* 42 (2016), 23–30.
- [4] G. Brunnett, T. Schreiber, J. Braun, The geometry of optimal degree reduction of Bézier curves, *Comput. Aided Geom. Des.* 13 (1996), 773–788.
- [5] G.-D. Chen, G.-J. Wang, Optimal degree reduction of Bézier curves with constraints of end-points continuity, *Comput. Aided Geom. Des.* 19 (2002), 365–377.
- [6] M. Eck, Least squares degree reduction of Bézier curves, *Comput. Aided Des.* 27 (1995), 845–851.
- [7] G.E. Farin, *Curves and Surfaces for Computer-Aided Geometric Design. A Practical Guide*, fifth edition, Academic Press, Boston, 2002.
- [8] P. Gospodarczyk, Degree reduction of Bézier curves with restricted control points area, *Comput. Aided Des.* 62 (2015), 143–151.
- [9] P. Gospodarczyk, S. Lewanowicz, P. Woźny, $G^{k,l}$ -constrained multi-degree reduction of Bézier curves, *Numer. Algor.* 71 (2016), 121–137.
- [10] B. Jüttler, The dual basis functions for the Bernstein polynomials, *Adv. Comput. Math.* 8 (1998), 345–352.
- [11] B.-G. Lee, Y. Park, J. Yoo, Application of Legendre-Bernstein basis transformations to degree elevation and degree reduction, *Comput. Aided Geom. Des.* 19 (2002), 709–718.
- [12] S. Lewanowicz, P. Woźny, Bézier representation of the constrained dual Bernstein polynomials, *Appl. Math. Comp.* 218 (2011), 4580–4586.
- [13] L. Lu, Explicit G^2 -constrained degree reduction of Bézier curves by quadratic optimization, *J. Comput. Appl. Math.* 253 (2013), 80–88.
- [14] L. Lu, Gram matrix of Bernstein basis: Properties and applications, *J. Comput. Appl. Math.* 280 (2015), 37–41.
- [15] L. Lu, G. Wang, Optimal degree reduction of Bézier curves with G^2 -continuity, *Comput. Aided Geom. Des.* 23 (2006), 673–683.
- [16] L. Lu, G. Wang, Application of Chebyshev II-Bernstein basis transformations to degree reduction of Bézier curves, *J. Comput. Appl. Math.* 221 (2008), 52–65.
- [17] A. Rababah, B.-G. Lee, J. Yoo, A simple matrix form for degree reduction of Bézier curves using Chebyshev-Bernstein basis transformations, *Appl. Math. Comp.* 181 (2006), 310–318.
- [18] A. Rababah, S. Mann, Iterative process for G^2 multi-degree reduction of Bézier curves, *Appl. Math. Comp.* 217 (2011), 8126–8133.
- [19] A. Rababah, S. Mann, Linear methods for G^1 , G^2 , and G^3 -multi-degree reduction of Bézier curves, *Comput. Aided Des.* 45 (2013), 405–414.
- [20] P. Woźny, S. Lewanowicz, Multi-degree reduction of Bézier curves with constraints, using dual Bernstein basis polynomials, *Comput. Aided Geom. Des.* 26 (2009), 566–579.